**Homework3**

**刘洋 3016218175**

*printPrimes()函数源码：  
public static void printPrimes (int n)*

*{*

*int curPrime;*

*int numPrimes;*

*boolean isPrime;*

*int [] primes = new int [MAXPRIMES];*

*primes [0] = 2;*

*numPrimes = 1;*

*curPrime = 2;*

*while (numPrimes < n)*

*{*

*curPrime++;*

*isPrime = true;*

*for (int i = 0; i <= numPrimes-1; i++)*

*{*

*if (curPrime%primes[i]==0)*

*{*

*isPrime = false;*

*break;*

*}*

*}*

*if (isPrime)*

*{*

*primes[numPrimes] = curPrime;*

*numPrimes++;*

*}*

*}*

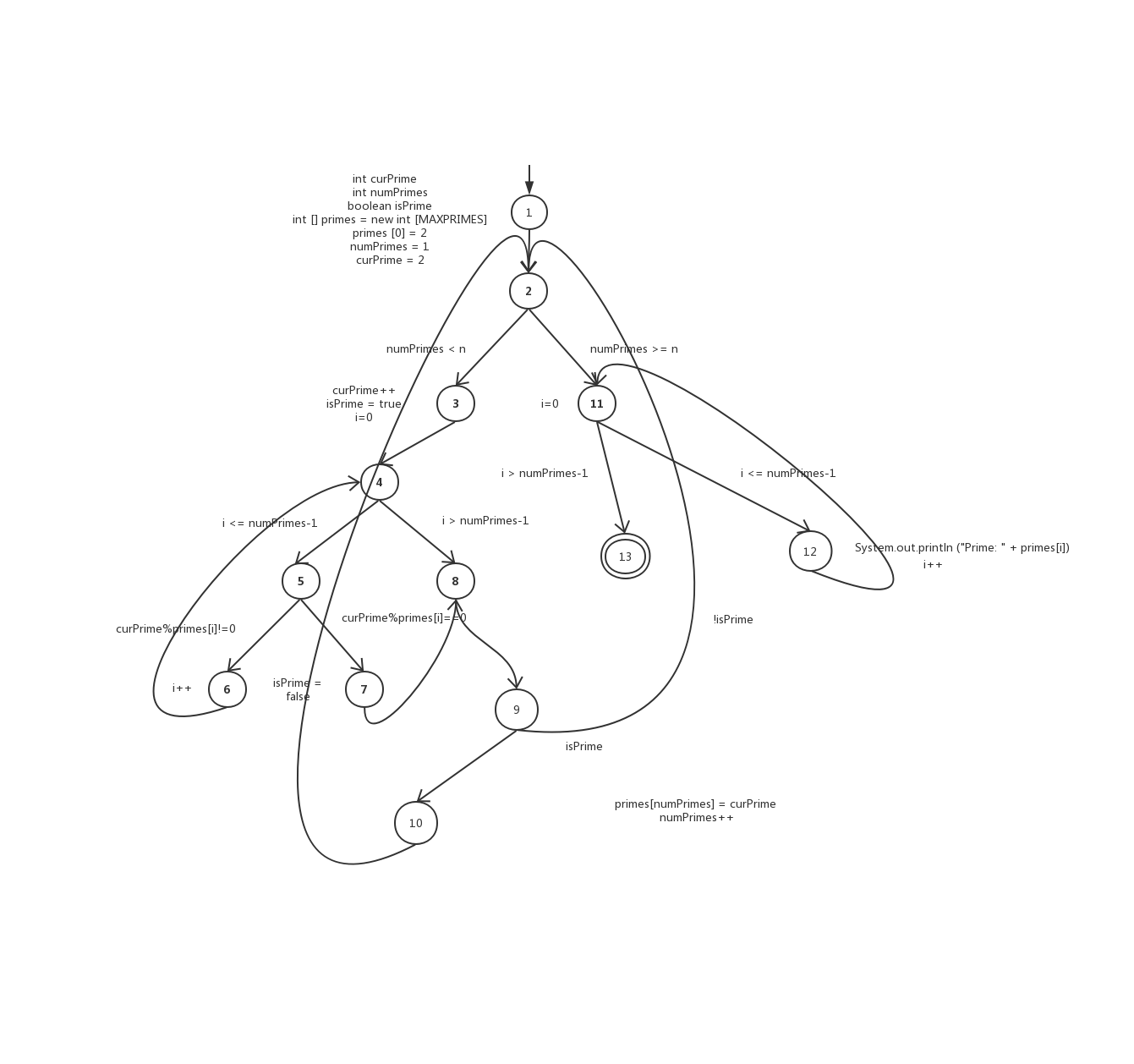
*for (int i = 0; i <= numPrimes-1; i++)*

*{*

*System.out.println ("Prime: " + primes[i]);*

*}*

1. 控制流图



1. Primes[]数组越界。例如，当MAXPRIMES = 2时，n=5就会导致primes[]数组越界，而n=3不会。
2. 当n=1时，满足条件
3. 节点覆盖：

TR={1,2,3,4,5,6,7,8,9,10,11,12,13}

边覆盖：

TR={(1,2),(2,3),(3,4),(4,5),(4,8),(5,6),(5,7),(6,4),(7,8),(8,9),(9,10),(9,2),(10,2),(2,11),(11,13),(11,12),(12,11)}

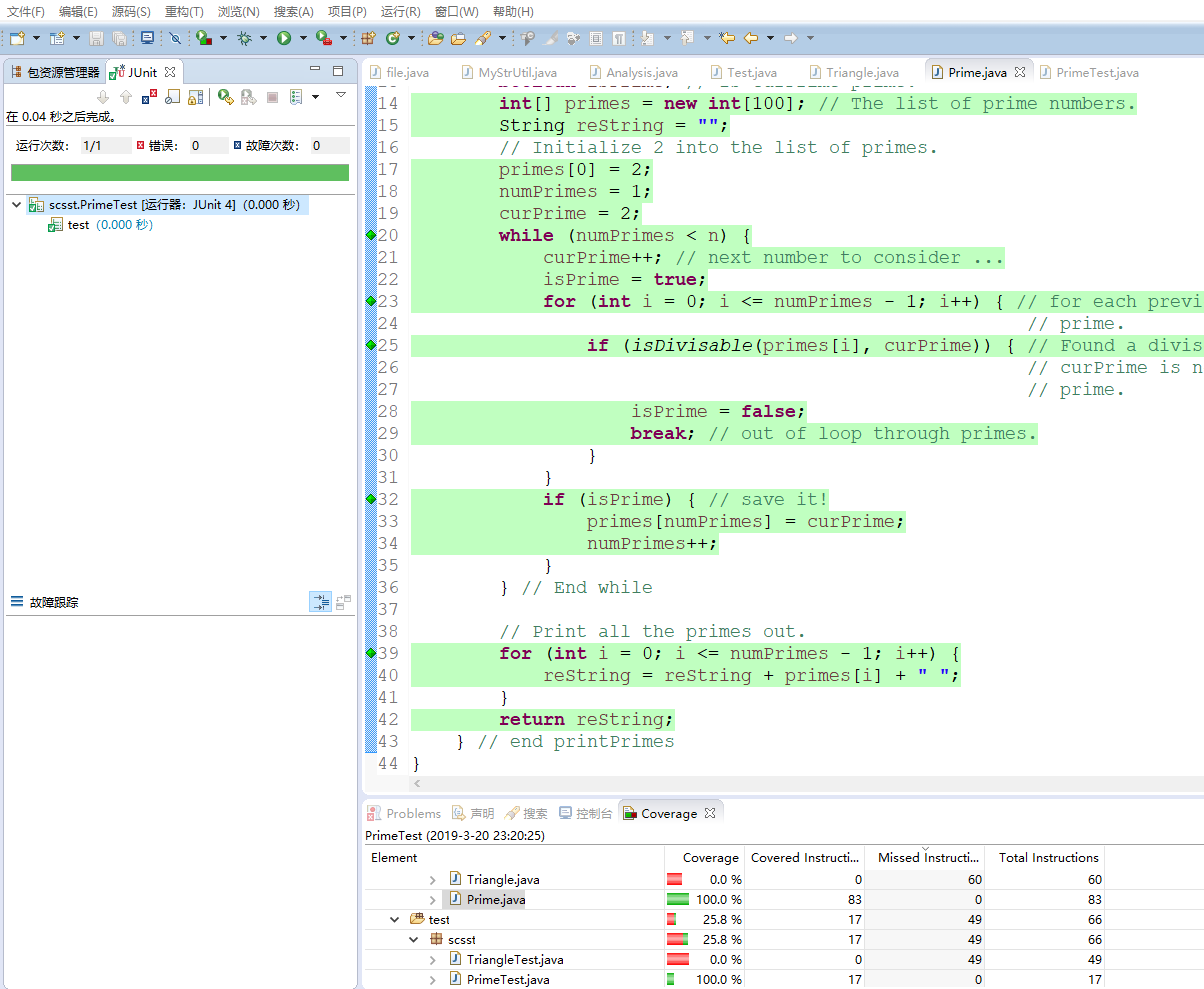
主路径覆盖：

TR={(1,2,3,4,5,6),(1,2,3,4,5,7,8,9,10),(1,2,3,4,5,7,8,9),(1,2,3,4,8,9,10),(1,2,3,4,8,9),(1,2,11,13),(1,2,11,12),(3,4,5,6),(3,4,5,7,8,9,10,2,11,13),(3,4,5,7,8,9,2,11,13),(3,4,5,7,8,9,2,11,12),(3,4,5,7,8,9,10,2,11,12),(3,4,8,9,10,2,11,13),(3,4,8,9,10,2,11,12),(3,4,8,9,2,11,13),(3,4,8,9,2,11,12),(4,5,6,4),(11,12,11),(12,11,13),(2,3,4,5,7,8,9,10,2),(2,3,4,5,7,8,9,2),(2,3,4,8,9,10,2),(2,3,4,8,9,2)}

基于Junit及Eclemma实现一个主路径覆盖测试

令n=3进行测试

结果如图：



其中本次使用的文件Prime.java PrimeTest.java

100%覆盖

测试代码：

